US 20180314646A1

(19) **United States**

(12) **Patent Application Publication**

Xu et al.

(10) **Pub. No.: US 2018/0314646 A1**

(43) **Pub. Date: Nov. 1, 2018**

(54) **CACHE MANAGEMENT METHOD, CACHE CONTROLLER, AND COMPUTER SYSTEM**

(71) Applicant: **Huawei Technologies Co., Ltd.**, Shenzhen (CN)

(72) Inventors: **Jun Xu**, Hangzhou (CN); **Yongbing Huang**, Beijing (CN); **Yuangang Wang**, Shenzhen (CN)

(21) Appl. No.: **16/028,265**

(22) Filed: **Jul. 5, 2018**

Related U.S. Application Data

(63) Continuation of application No. PCT/CN2016/070230, filed on Jan. 6, 2016.

Publication Classification

(51) **Int. Cl.**

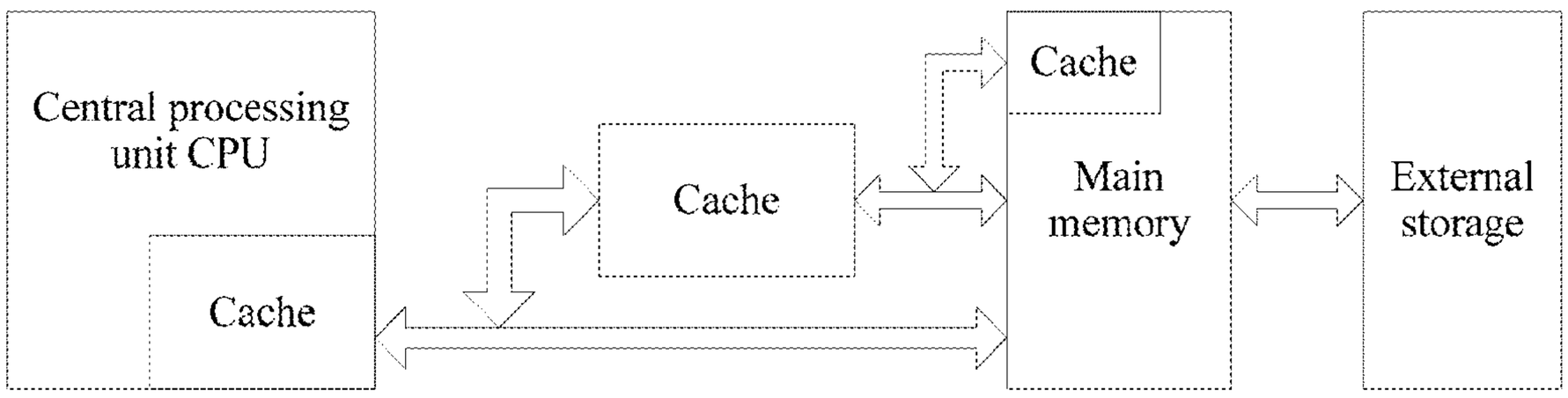
G06F 12/121	(2006.01)
G06F 12/0871	(2006.01)
G06F 12/0804	(2006.01)

(52) **U.S. Cl.**

CPC ***G06F 12/121*** (2013.01); ***G06F 12/0804*** (2013.01); ***G06F 12/0871*** (2013.01)

(57) **ABSTRACT**

A cache management method, a cache controller, and a computer system are provided. In the method, the cache controller obtains an operation instruction; when a destination address in the operation instruction hits no cache line cache line in a cache of the computer system, and the cache includes no idle cache line, the cache controller selects a to-be-replaced cache line from a replacement set, where the replacement set includes at least two cache lines; and the cache controller eliminates the to-be-replaced cache line from the cache, and stores, in the cache, a cache line obtained from the destination address. According to the cache management method, system overheads of cache line replacement can be reduced, and cache line replacement efficiency can be improved.



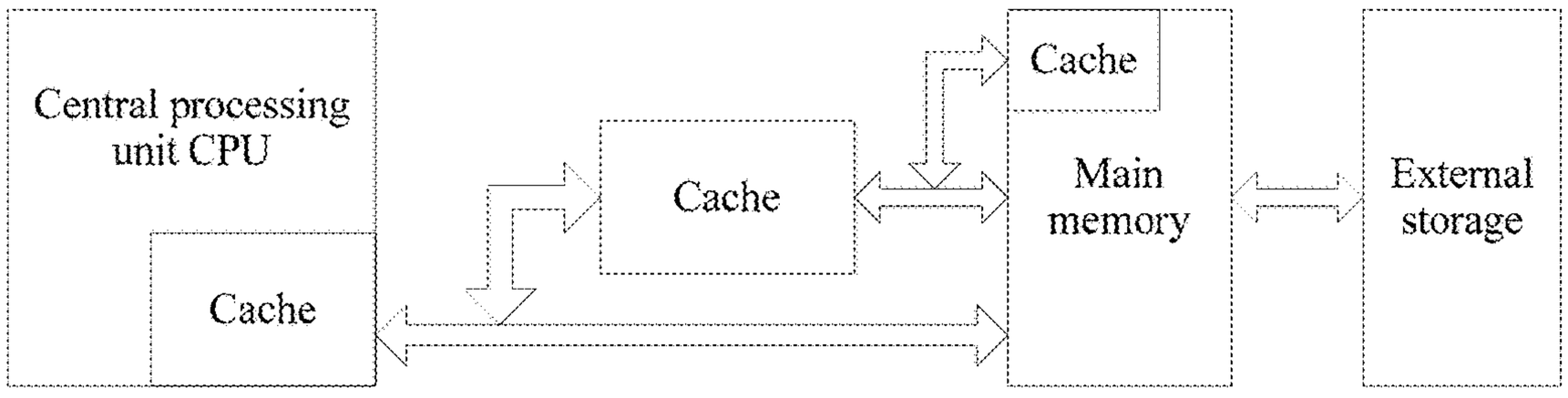

Central processing unit CPU
Cache
Cache
Cache
Main memory
External storage

FIG. 1

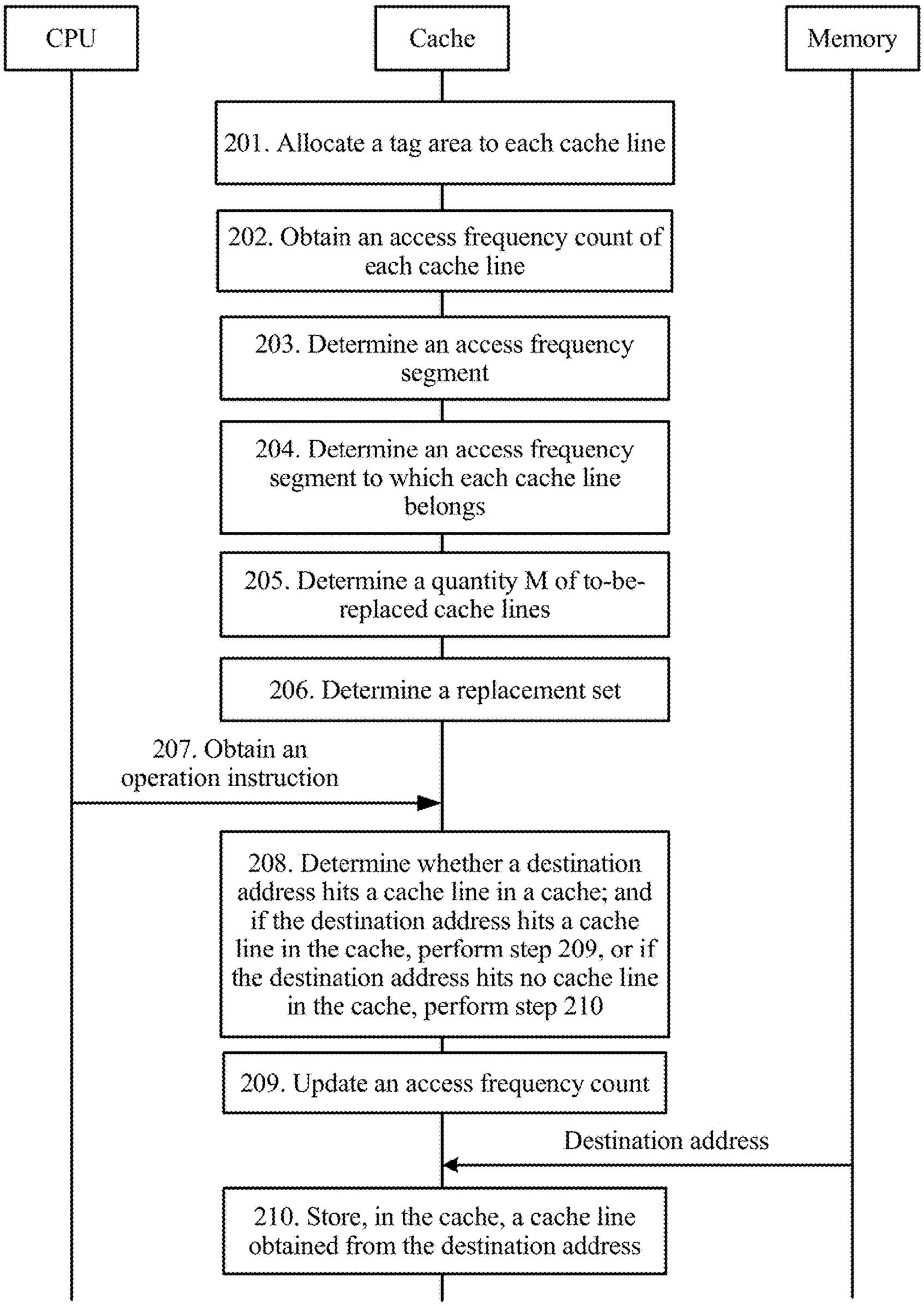

CPU
Cache
Memory
201. Allocate a tag area to each cache line
202. Obtain an access frequency count of each cache line
203. Determine an access frequency segment
204. Determine an access frequency segment to which each cache line belongs
205. Determine a quantity M of to-be-replaced cache lines
206. Determine a replacement set
207. Obtain an operation instruction
208. Determine whether a destination address hits a cache line in a cache; and if the destination address hits a cache line in the cache, perform step 209, or if the destination address hits no cache line in the cache, perform step 210
209. Update an access frequency count
Destination address
210. Store, in the cache, a cache line obtained from the destination address

FIG. 2

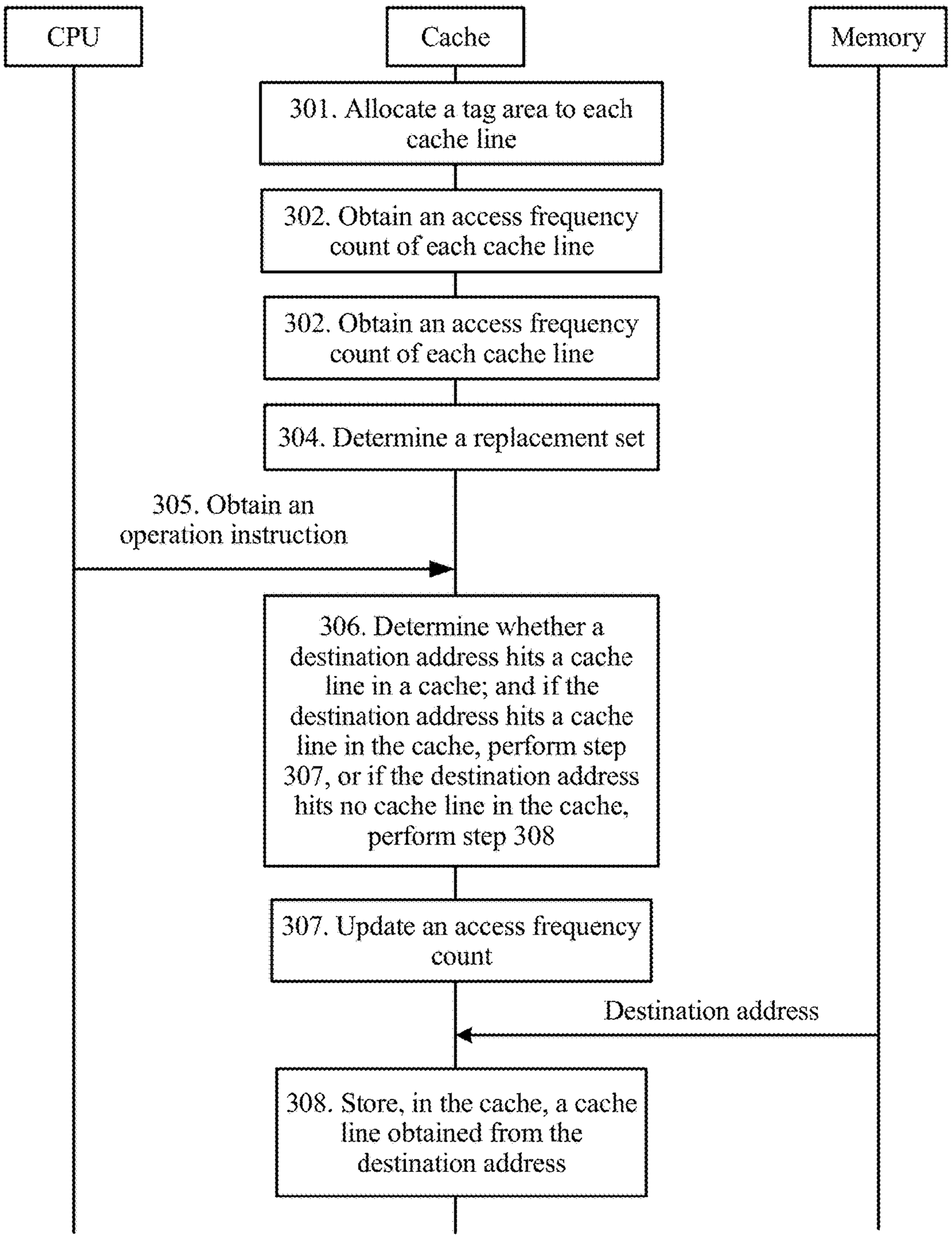

CPU
Cache
Memory
301. Allocate a tag area to each cache line
302. Obtain an access frequency count of each cache line
304. Determine a replacement set
305. Obtain an operation instruction
306. Determine whether a destination address hits a cache line in a cache; and if the destination address hits a cache line in the cache, perform step 307, or if the destination address hits no cache line in the cache, perform step 308
307. Update an access frequency count
Destination address
308. Store, in the cache, a cache line obtained from the destination address

FIG. 3

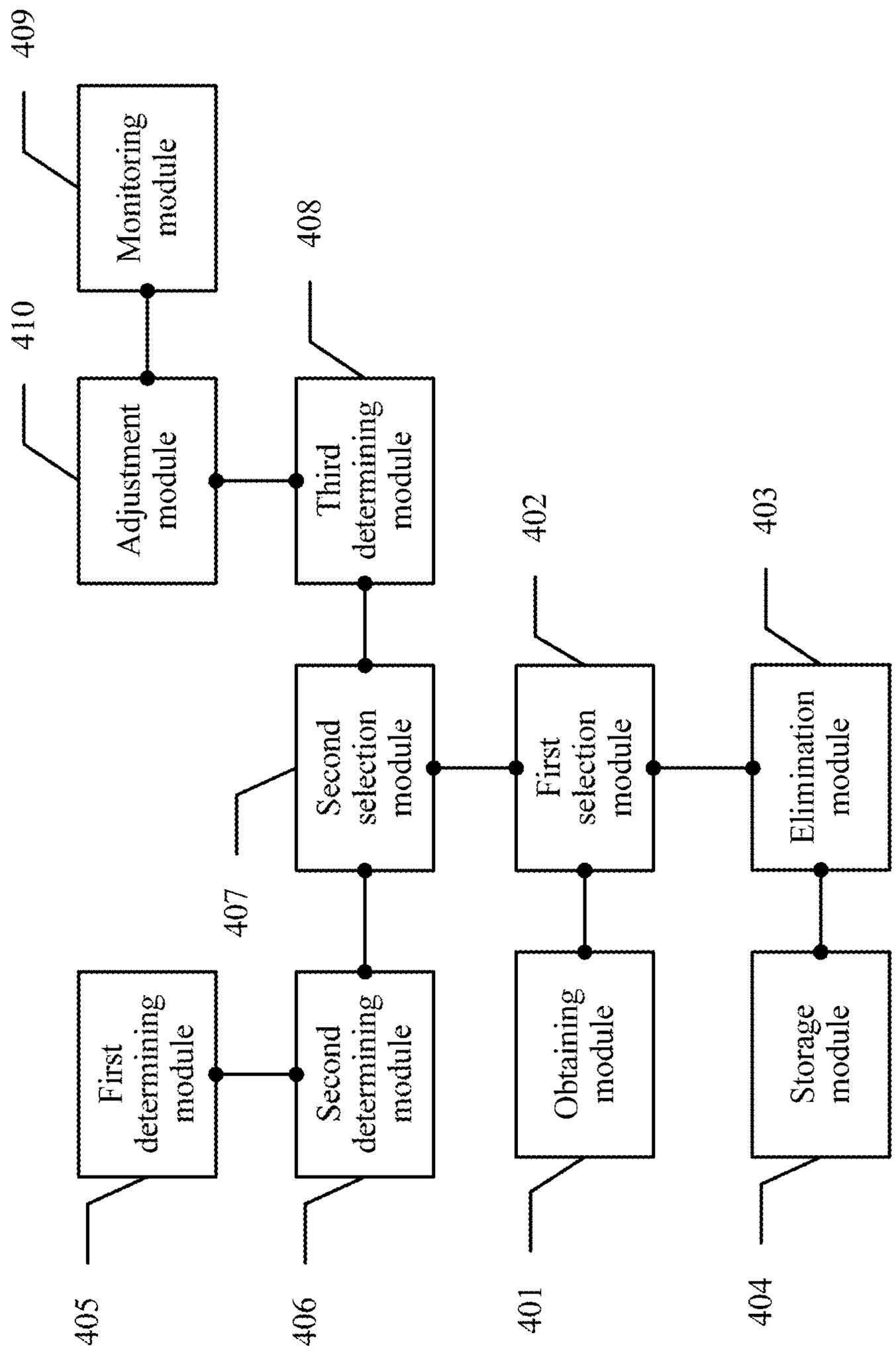

409
Monitoring module
410
Adjustment module
408
Third determining module
402
First selection module
403
Elimination module
407
Second selection module
405
First determining module
406
Second determining module
401
Obtaining module
404
Storage module

FIG. 4

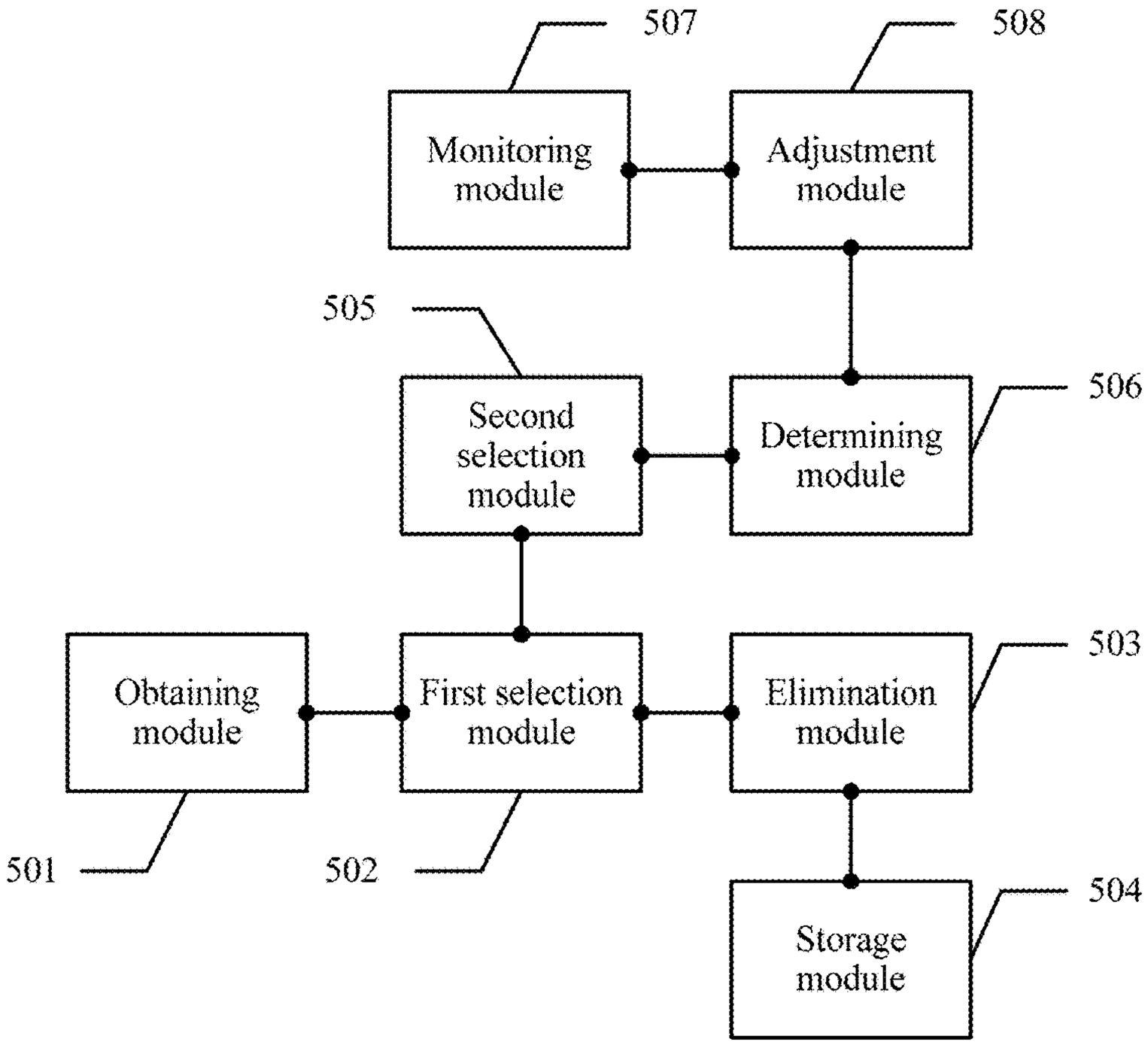

507
508
Monitoring module
Adjustment module
505
Second selection module
Determining module
506
Obtaining module
First selection module
Elimination module
503
501
502
Storage module
504

FIG. 5

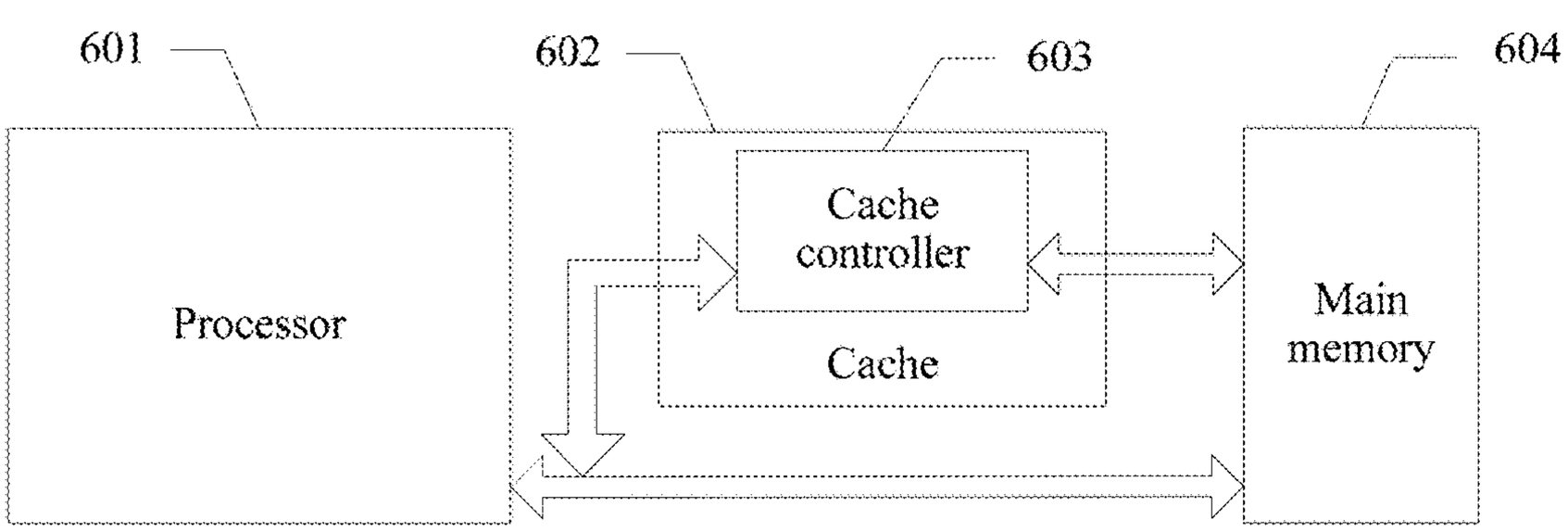

601
602
603
604
Processor
Cache controller
Cache
Main memory

FIG. 6

CACHE MANAGEMENT METHOD, CACHE CONTROLLER, AND COMPUTER SYSTEM

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application is a continuation of International Application No. PCT/CN2016/070230 filed on Jan. 6, 2016, the disclosure of which is hereby incorporated by reference in its entirety.

TECHNICAL FIELD

[0002] The present disclosure relates to the communications field, and in particular, to a cache management method, a cache controller, and a computer system.

BACKGROUND

[0003] A main idea of a cache technology is to place frequently-used data from a storage medium (for example, a disk) into a cache. Because a read/write speed of a cache is much higher than that of a disk, access efficiency of an entire system can be improved.

[0004] Generally, a cache has relatively good performance but a limited capacity, and can cache only some accessed data. It is necessary to try to hit, in the cache, a file related to a read/write operation that is about to occur, so as to allow better cache efficiency. There are two main aspects of cache management: how to select data to be cached in a cache (that is, cache insertion) and how to select to-be-replaced cache data (that is, cache replacement).

[0005] The prior art provides a cache management algorithm that is based on a statistical count, to add one counter for each cache line. The counter is used to count a quantity of access requests for accessing a cache line within a time interval, that is, is used to collect statistics on an access count of the cache line. In the prior art, two cache line replacement algorithms are implemented based on an access counter. In a first algorithm, a time interval is defined as a time interval between two consecutive times of accessing a cache line. After a value of the access counter exceeds a threshold Δthd, the cache line is used as a to-be-replaced target. In a second algorithm, a time interval is defined as a time interval starting from time when a cache line is added to a cache to current time. Likewise, after the value of the access counter exceeds a threshold LT, the cache line is used as a to-be-replaced target.

[0006] In the prior art, if data that currently needs to be read/written misses a cache line in a cache, a controller extracts the data from a storage medium and uses the data as a new cache line ready to put into the cache. In this case, if a quantity of cache lines stored in the cache has reached an upper limit, a cache line needs to be deleted from original cache lines.

[0007] The controller traverses counters of all original cache lines to learn a quantity of access requests of each cache line or an access time interval of each cache line, so as to select a cache line with a minimum quantity of access requests or a longest access time interval, and delete the cache line while putting a new cache line into the cache.

[0008] Therefore, in the prior art, in a cache line replacement process, a controller needs to traverse all original cache lines, and select a to-be-replaced cache line according to a calculation rule. Consequently, relatively large system overheads are caused, and cache line replacement efficiency is affected.

SUMMARY

[0009] Embodiments of the present disclosure provide a cache management method, a cache controller, and a computer system, to effectively reduce system overheads of cache line replacement, and improve cache line replacement efficiency.

[0010] A first aspect of the embodiments of the present disclosure provides a cache management method applied to a computer system. In this method, a cache controller obtains an operation instruction sent by a CPU, the operation instruction carries a destination address, and the destination address is an address that is in a memory and that is to be accessed by the CPU. In this case, if the cache controller does not find, in any cache line cache line in a cache of the computer system, a cache line that matches the destination address, that is, the destination address misses a cache line, and there is no idle cache line in the cache, the cache controller selects a to-be-replaced cache line from a pre-obtained replacement set for replacement. It should be noted herein that the replacement set includes at least two cache lines. The cache controller eliminates the selected to-be-replaced cache line from the cache. Then, the cache controller stores, in the cache, a cache line obtained from the destination address, to complete cache line replacement.

[0011] In the prior art, each time the destination address misses a cache line, the cache controller needs to traverse cache lines to find a to-be-replaced cache line; however, in the embodiments of the present disclosure, the cache controller can directly select a to-be-replaced cache line from the replacement set for replacement when the destination address misses a cache line. This effectively reduces system overheads of cache line replacement, and improves efficiency.

[0012] In a possible design, before the cache controller selects a to-be-replaced cache line from the replacement set, the cache controller may pre-obtain an access frequency count of each cache line in the cache, and divide an access frequency count range of a cache line into multiple access frequency segments according to a preset division policy. After determining, according to the access frequency count of each cache line, an access frequency segment to which each cache line belongs, the cache controller may select, from cache lines corresponding to the multiple access frequency segments, a to-be-replaced cache line according to a quantity M of to-be-replaced cache lines, to obtain the replacement set, and M is an integer that is not less than 2.

[0013] Optionally, in the embodiments of the present disclosure, instead of obtaining the multiple access frequency segments by means of division, the cache controller may directly select M to-be-replaced cache lines from cache lines in the cache to obtain the replacement set.

[0014] In the embodiments of the present disclosure, the cache controller divides the cache lines into multiple segments. This facilitates to-be-replaced cache line selection performed by the cache controller in unit of segment, and avoids a need to compare access frequency counts of cache lines on a large scale one by one during the to-be-replaced cache line selection, thereby reducing work overheads.

[0015] In a possible design, the selecting, by the cache controller from cache lines corresponding to the multiple

access frequency segments, a to-be-replaced cache line according to a quantity M of to-be-replaced cache lines includes: successively selecting, by the cache controller in ascending order of access frequency count ranges corresponding to the multiple access frequency segments, a cache line that belongs to each access frequency segment until a quantity of selected cache lines is equal to M.

[0016] In the embodiments of the present disclosure, a smaller access frequency count of a cache line represents a smaller quantity of times that the cache line is accessed, and later time at which the cache line is last accessed. Therefore, the cache line can be first replaced. The cache line successively selects, in ascending order of the access frequency count ranges, the cache line that belongs to each access frequency segment, so as to select an optimal replacement set.

[0017] In a possible design, the quantity M of to-be-replaced cache lines is determined according to an elimination ratio R and a total quantity of cache lines in the cache, and M is a product of the elimination ratio R and the total quantity of cache lines.

[0018] In another possible design, the cache controller may further periodically monitor an elimination frequency parameter, and the elimination frequency parameter includes at least one of a miss rate of each cache line in the cache or a traversal frequency of each cache line in the cache. When the elimination frequency parameter exceeds a first threshold, the cache controller adjusts the elimination ratio R to an elimination ratio R**1**, and the elimination ratio R**1** is greater than the elimination ratio R. When the elimination frequency parameter is less than a second threshold, the cache controller adjusts the elimination ratio R to an elimination ratio R**2**, the elimination ratio R**2** is less than the elimination ratio R, and the second threshold herein is less than the first threshold.

[0019] In the embodiments of the present disclosure, the cache controller dynamically adjusts the elimination ratio R according to a traversal rate of a cache line and a miss rate of the cache line, that is, a high miss rate, so that when the cache line has a high miss rate and is frequently traversed, a quantity of to-be-replaced cache lines in the replacement set is increased by raising the elimination ratio R. Alternatively, the cache controller may reduce the elimination ratio R when the cache line has a low miss rate and is less traversed, to reduce the quantity of to-be-replaced cache lines in the replacement set, thereby avoiding overheads caused by frequently selecting cache lines to generate the replacement set.

[0020] In a possible design, the selecting, by the cache controller, a to-be-replaced cache line from a replacement set includes: selecting, by the cache controller, a to-be-replaced cache line in ascending order of access frequency counts of cache lines in the replacement set, to achieve optimal cache line utilization.

[0021] In another possible design, after obtaining the replacement set, the cache controller may further monitor an access frequency count of a cache line that belongs to the replacement set, and eliminate, from the replacement set, a cache line whose access frequency count is greater than a third threshold within a preset time period. In other words, after obtaining the replacement set, the cache controller may continue to monitor a quantity of times that the cache line in the replacement set is hit. If an access frequency count of a cache line in the replacement set is greater than the third threshold within the preset time period, the cache controller may consider that the cache line is hit by a destination address in a subsequently received operation instruction, and therefore the cache line does not have to be eliminated. In this manner, the cache controller can update a cache line in the replacement set, to prevent some accessed cache lines from being incorrectly eliminated.

[0022] In a possible design, the cache controller may obtain the access frequency count of each cache line according to an access count of each cache line. Alternatively, the cache controller may obtain the access frequency count of each cache line according to an access count and access time of each cache line.

[0023] A second aspect of the present disclosure provides a cache controller. The cache controller includes a module configured to execute the method described in the first aspect and the various possible designs of the first aspect.

[0024] A third aspect of the present disclosure provides a computer system. The computer system includes a processor, a cache, and a cache controller. The processor is configured to send an operation instruction. The cache controller is configured to execute the method described in the first aspect and the various possible designs of the first aspect.

[0025] In the embodiments of the present disclosure, the cache controller obtains the operation instruction. The operation instruction carries the destination address, and the destination address is the address that is in the memory and that is to be accessed in the operation instruction. When the destination address hits no cache line cache line in the cache of the computer system, and the cache includes no idle cache line, the cache controller selects the to-be-replaced cache line from the replacement set. The replacement set includes at least two cache lines. The cache controller eliminates the to-be-replaced cache line from the cache; and the cache controller stores, in the cache, the cache line obtained from the destination address. Therefore, in a cache line replacement process, the cache controller only needs to select the to-be-replaced cache line from the replacement set. The replacement set is pre-selected, and this effectively improves cache line replacement efficiency.

[0026] According to a fourth aspect of the present disclosure, this application provides a computer program product, including a computer readable storage medium that stores program code. An instruction included in the program code is used to execute at least one method described in the first aspect and the various possible designs of the first aspect.

BRIEF DESCRIPTION OF THE DRAWINGS

[0027] To describe the technical solutions in the embodiments of the present disclosure more clearly, the following briefly describes the accompanying drawings required for describing the embodiments. Apparently, the accompanying drawings in the following description show merely some embodiments of the present disclosure.

[0028] FIG. **1** is a framework diagram of a computer system according to an embodiment of the present disclosure;

[0029] FIG. **2** is a schematic diagram of a cache management method according to an embodiment of the present disclosure;

[0030] FIG. **3** is another schematic diagram of a cache management method according to an embodiment of the present disclosure;

[0031] FIG. **4** is a schematic diagram of a cache controller according to an embodiment of the present disclosure;

[0032] FIG. **5** is a schematic diagram of a cache controller according to an embodiment of the present disclosure; and

[0033] FIG. **6** is a schematic architecture diagram of another computer system according to an embodiment of the present disclosure.

DETAILED DESCRIPTION

[0034] Embodiments of the present disclosure provide a cache management method, a cache controller, and a computer system, to effectively reduce system overheads of cache line replacement, and improve cache line replacement efficiency.

[0035] In the specification, claims, and accompanying drawings of the present disclosure, the terms "first", "second", "third", "fourth", and so on (if existent) are intended to distinguish between similar objects but do not necessarily indicate order or sequence. It should be understood that the data termed in such a way are interchangeable in proper circumstances so that the embodiments of the present disclosure described herein can be implemented in other orders than the order illustrated or described herein.

[0036] Referring to FIG. **1**, FIG. **1** is an architecture diagram of a computer system according to an embodiment of the present disclosure. As shown in FIG. **1**, a cache is a small-capacity memory between a CPU and a main memory, and an access speed of the cache is quicker than that of the main memory, and is close to that of the CPU. The cache can provide an instruction and data for the CPU at a high speed, to improve an execution speed of a program. A cache technology is an important technology for resolving a problem of a speed mismatch between the CPU and the main memory.

[0037] A cache is a buffer memory of a main memory, and includes a high-speed static random access memory (SRAM). The cache may be built into a central processing unit CPU, or may be built into a memory, or the cache may be an independent entity externally existing between a CPU and a memory. All cache control logic is implemented by an internal cache controller. As semiconductor device integration constantly improves, currently a multi-level cache system with at least two levels of caches already emerges, such as an L1 cache (level 1 cache), an L2 cache (level 2 cache), and an L3 cache (level 3 cache).

[0038] Currently, a cache capacity is very small, content stored in the cache is only a subset of main memory content, and a data exchange between the CPU and the cache is in unit of word, but a data exchange between the cache and the main memory is in unit of cache line. One cache line includes multiple fixed-length words. When the CPU reads a word in the main memory, a main memory address of the word is sent to the cache and the main memory. In this case, cache control logic determines, according to the address, whether the word currently exists in the cache. If the word currently exists in the cache, the word is immediately transmitted from the cache to the CPU, or if the word currently does not exist in the cache, the word is read from the main memory in a main memory read period to send to the CPU, and an entire cache line that includes this word is read from the main memory to send to the cache. In addition, a cache line in the cache is replaced by using a replacement policy, and a replacement algorithm is implemented by a cache management logic circuit.

[0039] In the prior art, during cache line replacement, a controller traverses all original cache lines to learn a quantity of access requests of each cache line or an access time interval of each cache line, so as to select a cache line with a minimum quantity of access requests or a longest access time interval, and delete the cache line while putting a new cache line into a cache. In the prior art, system overheads of the cache line replacement are relatively large, and efficiency is not high.

[0040] An embodiment of the present disclosure provides a cache management method. In the cache management method provided in this embodiment of the present disclosure, in each cache line replacement process, there is no need to traverse all original cache lines, and there is no need to select a to-be-replaced cache line by means of calculation. This effectively reduces system overheads, and improves cache line replacement efficiency. For ease of understanding, the following describes, in detail by using a procedure in this embodiment of the present disclosure, a cache management method of the computer system shown in FIG. **1**. It may be understood that the cache management method in FIG. **2** may be applied to the cache shown in FIG. **1**. Referring to FIG. **2**, an embodiment of the cache management method in this embodiment of the present disclosure includes the following steps.

[0041] **201**. A cache controller allocates a tag area to each cache line.

[0042] In this embodiment, the cache controller may allocate a data area and a tag area to each cache line. The data area may be used to store actual file data, and the tag area may be used to store a file number, an offset, length information, or data address information that is corresponding to the data area. Details are not described herein.

[0043] It should be noted that a storage space size of each area in the data area may be dynamically changed or may be static. In actual application, the cache controller may record a valid character length of the data area in the tag area, to dynamically change a storage space size of the data area. Details are not described herein.

[0044] In this embodiment, the cache controller may add an access count counter and an access time counter for each tag area. The access count counter may be used to collect statistics on a quantity of times that a cache line is accessed, and the access time counter may be used to record time at which the cache line is last accessed. The time may be identified by an absolute time value, or may be identified by a quantity of time periods. Details are not limited herein.

[0045] It should be noted that the cache controller may collect statistics on an access count for each cache line, or may simultaneously collect statistics on the access count and access time for each cache line. Details are not limited herein.

[0046] In this embodiment of the present disclosure, a cache may be built into a CPU, or may be built into a memory, or may be an independent entity. The entity may include a cache controller and a cache line, and composition of the cache line may include a static random access memory (SRAM). Details are not limited herein.

[0047] **202**. The cache controller obtains an access frequency count of each cache line.

[0048] In this embodiment, the cache controller may obtain the access frequency count of each cache line by detecting the tag area of each cache line. The access frequency count may include at least one of an access count or

access time. Details are not limited herein. For example, in a case, the access frequency count of each cache line may be obtained according to an access count of each cache line. In this case, the access count of a cache line may be used as an access frequency count of the cache line. In another case, alternatively, the access frequency count of each cache line may be obtained according to an access count and access time of each cache line. For example, in actual application, an implementation in which the cache controller may adjust the access frequency count by using the access time may be as follows: The cache controller maps the access time to a time slice, where one time slice includes a short time period, calculates a difference between a current time slice and a time slice in which last access time of the cache line is located, and adjusts the access frequency count according to the time slice difference (for example, Access frequency count=Current access count−Time slice difference×Delta; the Delta value may be statically set according to an empirical value).

[0049] It should be noted that the cache controller may update the access count or the access time of each cache line after the cache line is accessed. In actual application, an access time update manner may be recording a current access time, or may be returning the access time counter to zero and restarting counting. Details are not limited herein.

[0050] **203**. The cache controller determines an access frequency segment.

[0051] In this embodiment, the cache controller may determine multiple access frequency segments according to the access frequency count of each cache line in the cache and a preset division policy, and each access frequency segment may be corresponding to a different access frequency count range.

[0052] In this embodiment, the preset division policy may be as follows: The cache controller first determines a parameter upper limit value and a parameter lower limit value based on the access frequency count of each cache line. The parameter lower limit value may be 0, and the parameter upper limit value may be access_times_max. It should be noted that, in actual application, access_times_max_may be a value preset by a user according to experience, or may be obtained by the cache controller by collecting statistics on the access frequency count of each cache line. Details are not limited herein.

[0053] It should be noted that, in this embodiment, if an access frequency count of a cache line is greater than access_times_max, the cache controller may determine that the cache line does not need to be replaced.

[0054] According to the preset division policy in this embodiment, the cache controller may divide a range from the parameter lower limit value 0 to the parameter upper limit value access_times_max into N access frequency segments after determining the parameter upper limit value and the parameter lower limit value. The value N may be a value entered by the user in advance. It may be understood that, in actual application, a size of each access frequency segment may be a fixed value access_times_max/N, or may be a random value. The random value may be a value randomly generated by the cache controller, or may be a value entered by the user in advance. Details are not limited herein.

[0055] It should be noted that, in actual application, a sum of sizes of all access frequency segments is equal to access_times_max.

[0056] In this embodiment, the cache controller may further establish a data structure. The data structure is used to record a quantity of cache lines, in each access frequency segment, whose access frequency count belongs to the access frequency segment, or may establish a corresponding data structure for each access frequency segment. Details are not limited herein.

[0057] **204**. The cache controller determines an access frequency segment to which each cache line belongs.

[0058] In this embodiment, the cache controller may determine, by traversing cache lines, a target access frequency count range to which the access frequency count of each cache line belongs, and determine a target access frequency segment corresponding to the target access frequency count range to which each cache line belongs. It should be noted that the cache controller may record, by using the data structure, a quantity of cache lines included in each access frequency segment, to collect statistics on the quantity of cache lines in the access frequency segment.

[0059] In should be noted that, in actual application, each time determining that one cache line belongs to one access frequency segment, the cache controller may correspondingly adjust a data structure corresponding to the access frequency segment. It may be understood that, in the actual application, each time determining that one cache line belongs to one access frequency segment, the cache controller may increase a quantity recorded by the data structure corresponding to the access frequency segment by a value 1.

[0060] **205**. The cache controller determines a quantity M of to-be-replaced cache lines.

[0061] In this embodiment, the cache controller may preselect M to-be-replaced cache lines according to an elimination ratio R. The to-be-replaced cache lines may be cache lines with a small access frequency count, and the quantity M of to-be-replaced cache lines may be a product of the elimination ratio R and a total quantity of cache lines, and M is an integer not less than 2. It should be noted that, in actual application, the elimination ratio R may be a fixed value, or may be dynamically adjusted by the cache controller. Details are not limited herein.

[0062] In this embodiment, the cache controller may determine, by monitoring an elimination frequency parameter in real time, whether the elimination frequency parameter exceeds a first threshold; and if the elimination frequency parameter exceeds the first threshold, the cache controller may adjust the elimination ratio R to an elimination ratio R**1**, or if the elimination frequency parameter does not exceed the first threshold, the cache controller may further determine whether the elimination frequency parameter is less than a second threshold. If the elimination frequency parameter is less than the second threshold, the cache controller may adjust the elimination ratio R to an elimination ratio R**2**. It should be noted that if the elimination frequency parameter is not less than the second threshold, the cache controller may continue to monitor the elimination frequency parameter. The first threshold and the second threshold may be values preset by a user, and the first threshold is less than the second threshold.

[0063] It should be noted that, in actual application, the cache controller may increase the elimination ratio R by A to obtain the elimination ratio R**1**, or may decrease the elimination ratio R by A to obtain the elimination ratio R**2**. The value A may be a value preset by a user.

[0064] In this embodiment, the elimination frequency parameter may include at least one of a miss rate of each cache line in the cache or a traversal frequency of each cache line in the cache, and the cache controller may return to continue to monitor the elimination frequency parameter each time after adjusting the elimination ratio R.

[0065] **206**. The cache controller determines a replacement set.

[0066] In this embodiment, the cache controller may successively accumulate a cache line in each access frequency segment into the replacement set in ascending order of values corresponding to an access frequency count range until a quantity of cache lines in the replacement set is equal to M. It may be understood that, in actual application, the accumulating, by the cache controller, a cache line in each access frequency segment may include the following steps:

[0067] 1. The cache controller may determine an access frequency segment whose value corresponding to an access frequency count range is lowest in an access frequency segment group as an accumulative access frequency segment, and the access frequency segment group includes each access frequency segment.

[0068] 2. The cache controller may remove the accumulative access frequency segment from the access frequency segment group.

[0069] 3. The cache controller may accumulate a cache line in the accumulative access frequency segment into the replacement set.

[0070] 4. The cache controller may determine whether a quantity of candidate cache lines in the replacement set is equal to M; and if the quantity of candidate cache lines in the replacement set is equal to M, the cache controller may end a procedure, or if the quantity of candidate cache lines in the replacement set is less than M, the cache controller may determine whether a next access frequency segment to be accumulated into the replacement set is a critical access frequency segment. If the next access frequency segment to be accumulated into the replacement set is not a critical access frequency segment, the cache controller may repeatedly perform steps 1 to 4; or if the next access frequency segment to be accumulated into the replacement set is a critical access frequency segment, the cache controller may randomly select, from the critical access frequency segment, X cache lines to accumulate into the replacement set. X may be a difference between M and a quantity of cache lines that have been accumulated into the replacement set. It should be noted that the critical access frequency segment is such an access frequency segment that before an included cache line of the access frequency segment is accumulated into the replacement set, the quantity of candidate cache lines in the replacement set is less than M and after the included cache line of the access frequency segment is accumulated into the replacement set, the quantity of candidate cache lines in the replacement set is greater than M.

[0071] It should be noted that if an accumulative value in a current replacement set does not exceed M, the cache controller may continue to traverse access frequency segments until the accumulative value is greater than or equal to M.

[0072] It should be noted that, in actual application, the cache controller may first accumulate each access frequency segment into a set of to-be-eliminated range segments, and then may select, by traversing cache lines, a cache line that belongs to a to-be-eliminated access frequency segment. The cache controller may determine the replacement set by using the selected cache line as a cache line in the replacement set.

[0073] **207**. The cache controller obtains an operation instruction.

[0074] The cache controller may receive an operation instruction sent by a CPU. The operation instruction may carry a destination address, and the destination address may be an address that is in a memory and that is to be accessed in the operation instruction.

[0075] **208**. The cache controller determines whether a destination address hits a cache line in a cache; and if the destination address hits a cache line in the cache, the cache controller performs step **209**, or if the destination address hits no cache line in the cache, the cache controller performs step **210**.

[0076] When the CPU reads a word in a main memory, a main memory address of the word may be sent to the cache and the main memory. In this case, the cache controller may determine, according to the address, whether the word currently exists in the cache. It should be noted that, in actual application, the cache controller may first match a cache line outside the replacement set, or may first match a cache line in the replacement set. Details are not limited herein.

[0077] **209**. The cache controller updates an access frequency count.

[0078] In this embodiment, the cache controller may update an access frequency count of the cache line after the destination address hits a cache line in the cache. An update manner may be increasing a value of a counter for a tag area of the cache line.

[0079] In actual application, the cache controller may further monitor an access frequency count of a cache line in the replacement set in real time, and may eliminate, from the replacement set, a cache line whose access frequency count is greater than a third threshold within a preset time period. For example, the third threshold may be 1. In other words, after obtaining the replacement set, the cache controller may continue to monitor a quantity of times that the cache line in the replacement set is hit. If an access frequency count of a cache line in the replacement set is greater than the third threshold within a preset time period, the cache controller may consider that the cache line is hit by a destination address in a subsequently received operation instruction and may further be accessed, and therefore the cache line does not have to be eliminated. In this manner, the cache controller can update a cache line in the replacement set, to prevent some accessed cache lines from being incorrectly eliminated.

[0080] It should be noted that the cache controller completes the step procedure after updating the access frequency count of the cache line.

[0081] **210**. The cache controller stores, in the cache, a cache line obtained from the destination address.

[0082] In this embodiment, the cache controller may obtain the destination address from a storage medium after the destination address hits no cache line in the cache, and may write the cache line obtained from the destination address to replace a to-be-replaced cache line selected from the replacement set. It should be noted that, in actual application, the cache controller may first select, in ascending order of values corresponding to access frequency counts, a cache line with a small access frequency count from the replacement set as the to-be-replaced cache line.

[0083] It should be noted that the cache controller completes the step procedure after writing the destination address to replace the to-be-replaced cache line in the replacement set.

[0084] In a scenario of this embodiment of the present disclosure, the cache includes 10 cache lines, and the cache controller may obtain the access frequency count of each cache line. For details, refer to Table 1.

TABLE 1

	Cache line									
	A1	A2	A3	A4	A5	B1	B2	B3	B4	B5
Access frequency count	1	5	10	12	15	20	20	25	30	35

[0085] The cache controller may determine a replacement set according to the access frequency count of each cache line in Table 1, and a process thereof is as follows:

[0086] The cache controller determines that a lower limit value of the access frequency count is 0 and an upper limit value of the access frequency count is 30, and the cache controller determines that a cache line whose access frequency count is greater than the parameter upper limit 30 does not need to be eliminated.

[0087] The cache controller may uniformly divide access frequency counts 0 to 30 into three access frequency segments (that is, a range segment #1 is 0 to 10, a range segment #2 is 11 to 20, and a range segment #3 is 21 to 30).

[0088] The cache controller may obtain a quantity of cache lines in each access frequency segment by traversing cache lines. For details, refer to Table 2.

TABLE 2

	Range segment		
	Range segment #1	Range segment #2	Range segment #3
Access frequency count	0 to 10	11 to 20	21 to 30
Quantity of cache lines	3	4	2
Cache line	A1, A2, and A3	A4, A5, B1, and B2	B3 and B4

[0089] When the cache controller determines that the elimination ratio R is 50%, the cache controller may learn, by means of calculation, that the quantity M of to-be-replaced cache lines is 5 (M=Total quantity of cache lines× Elimination ratio R=10×50%=5). The cache controller may determine that cache lines A1, A2, and A3 in the range segment #1 may be to-be-replaced cache lines, and the cache controller may determine that the range segment #2 is a critical access frequency segment. A quantity of cache lines that need to be eliminated in the critical access frequency segment, that is, the range segment #2, is 2 (M−Quantity of cache lines in the range segment #1=5−3=2).

[0090] Referring to Table 1 and Table 2, the cache controller may accumulate, in ascending order of values corresponding to an access frequency count range, to-be-eliminated cache lines as a replacement set, that is, successively select the cache lines A1, A2, A3, A4, and A5 as the replacement set.

[0091] When the destination address hits no cache line in the cache, the cache controller may first select the to-be-replaced cache line A1 with a smallest access frequency count from the replacement set, and write the destination address read from the storage medium to replace the to-be-replaced cache line A1.

[0092] In this embodiment of the present disclosure, the cache controller obtains the operation instruction. The operation instruction carries the destination address, and the destination address is the address that is in the memory and that is to be accessed in the operation instruction. If the destination address hits no cache line cache line in the cache, and the cache includes no idle cache line, the cache controller selects the to-be-replaced cache line from the replacement set. The replacement set includes at least two cache lines. The cache controller eliminates the to-be-replaced cache line from the cache; and the cache controller stores, in the cache, the cache line obtained from the memory according to the destination address. Therefore, in a cache line replacement process, the cache controller only needs to select the to-be-replaced cache line from the replacement set. The replacement set is pre-selected, and this effectively improves cache line replacement efficiency.

[0093] Different from the embodiment shown in FIG. **2**, in an optional solution in an embodiment of the present disclosure, instead of obtaining the access frequency segments by means of division, the cache controller may directly determine the replacement set by using the access frequency count of each cache line.

[0094] Referring to FIG. **3**, another embodiment of a cache management method in this embodiment of the present disclosure includes the following steps.

[0095] Steps **301** and **302** in this embodiment are the same as steps **201** and **202** in the embodiment shown in FIG. **2**, and details are not described herein.

[0096] Step **303** in this embodiment is the same as step **205** in the embodiment shown in FIG. **2**, and details are not described herein again.

[0097] **304**. The cache controller determines a replacement set.

[0098] In this embodiment, the cache controller may successively accumulate each cache line into the replacement set in ascending order of values corresponding to an access frequency count range until a quantity of to-be-replaced cache lines in the replacement set is equal to M, and the cache line accumulated into the replacement set is a to-be-replaced cache line.

[0099] Steps **305** to **308** in this embodiment are the same as steps **207** to **210** in the embodiment shown in FIG. **2**, and details are not described herein again.

[0100] The foregoing describes the cache management method in this embodiment of the present disclosure, and the following describes a cache controller in an embodiment of the present disclosure. FIG. **4** describes an apparatus embodiment corresponding to the method embodiment in FIG. **2**. Referring to FIG. **4**, an embodiment of the cache controller in this embodiment of the present disclosure includes:

[0101] an obtaining module **401**, configured to obtain an operation instruction, where the operation instruction carries

a destination address, and the destination address is an address that is to be accessed in the operation instruction;

[0102] a first selection module **402**, configured to: when the destination address hits no cache line cache line in a cache of a computer system, and the cache includes no idle cache line, select a to-be-replaced cache line from a replacement set, where the replacement set includes at least two cache lines;

[0103] an elimination module **403**, configured to eliminate the to-be-replaced cache line from the cache; and

[0104] a storage module **404**, configured to store, in the cache, a cache line obtained from the destination address.

[0105] The cache controller in this embodiment further includes:

[0106] a first determining module **405**, configured to determine multiple access frequency segments according to an access frequency count of each cache line in the cache and a preset division policy, where each access frequency segment is corresponding to a different access frequency count range;

[0107] a second determining module **406**, configured to determine, according to the access frequency count of each cache line in the cache, an access frequency segment to which each cache line belongs; and

[0108] a second selection module **407**, configured to select, from cache lines corresponding to the multiple access frequency segments, a to-be-replaced cache line according to a quantity M of to-be-replaced cache lines, to obtain the replacement set, where M is an integer not less than 2.

[0109] In this embodiment, the second selection module **407** is configured to successively select, in ascending order of access frequency count ranges corresponding to the multiple access frequency segments, a cache line that belongs to each access frequency segment until a quantity of selected cache lines is equal to M.

[0110] The cache controller in this embodiment further includes:

[0111] a third determining module **408**, configured to determine the quantity M of to-be-replaced cache lines according to an elimination ratio R and a total quantity of cache lines in the cache, where M is a product of the elimination ratio R and the total quantity of cache lines.

[0112] The cache controller in this embodiment may further include:

[0113] a monitoring module **409**, configured to monitor an elimination frequency parameter, where the elimination frequency parameter includes at least one of a miss rate of each cache line in the cache or a traversal frequency of each cache line in the cache; and

[0114] an adjustment module **410**, configured to: when the elimination frequency parameter exceeds a first threshold, adjust the elimination ratio R to an elimination ratio R**1**, where the elimination ratio R**1** is greater than the elimination ratio R.

[0115] The adjustment module **410** is further configured to: when the elimination frequency parameter is less than a second threshold, adjust the elimination ratio R to an elimination ratio R**2**, where the elimination ratio R**2** is less than the elimination ratio R, and the second threshold is less than the first threshold.

[0116] The monitoring module **409** is configured to monitor an access frequency count of a cache line that belongs to the replacement set.

[0117] The elimination module **403** is configured to eliminate, from the replacement set, a cache line whose access frequency count is greater than a third threshold within a preset time period.

[0118] In this embodiment, the obtaining module **401** obtains the operation instruction. The operation instruction carries the destination address, and the destination address is the address that is to be accessed in the operation instruction. When the destination address hits no cache line cache line in the cache of the computer system, and the cache includes no idle cache line, the first selection module **402** selects the to-be-replaced cache line from the replacement set. The replacement set includes at least two cache lines. The elimination module **403** eliminates the to-be-replaced cache line from the cache. The storage module **404** stores, in the cache, the cache line obtained from the destination address. Therefore, in a cache line replacement process, the first selection module **402** only needs to select the to-be-replaced cache line from the replacement set. The replacement set is pre-selected, and this effectively improves cache line replacement efficiency.

[0119] The foregoing describes the apparatus embodiment corresponding to the method embodiment in FIG. **2**, and the following describes an apparatus embodiment corresponding to the method embodiment shown in FIG. **3**. Referring to FIG. **5**, another embodiment of a cache controller in this embodiment of the present disclosure includes:

[0120] an obtaining module **501**, configured to obtain an operation instruction, where the operation instruction carries a destination address, and the destination address is an address that is to be accessed in the operation instruction;

[0121] a first selection module **502**, configured to: when the destination address hits no cache line cache line in a cache of a computer system, and the cache includes no idle cache line, select a to-be-replaced cache line from a replacement set, where the replacement set includes at least two cache lines;

[0122] an elimination module **503**, configured to eliminate the to-be-replaced cache line from the cache; and

[0123] a storage module **504**, configured to store, in the cache, a cache line obtained from the destination address.

[0124] The cache controller in this embodiment further includes:

[0125] a second selection module **505**, configured to select, from cache lines in the cache, a to-be-replaced cache line according to a quantity M of to-be-replaced cache lines, to obtain the replacement set, where M is an integer not less than 2.

[0126] The second selection module **505** in this embodiment is configured to successively select a cache line in ascending order of access frequency count ranges corresponding to cache lines until a quantity of selected cache lines is equal to M.

[0127] The cache controller in this embodiment further includes:

[0128] a determining module **506**, configured to determine the quantity M of to-be-replaced cache lines according to an elimination ratio R and a total quantity of cache lines in the cache, where M is a product of the elimination ratio R and the total quantity of cache lines.

[0129] The cache controller in this embodiment may further include:

[0130] a monitoring module **507**, configured to monitor an elimination frequency parameter, where the elimination fre-

quency parameter includes at least one of a miss rate of each cache line in the cache or a traversal frequency of each cache line in the cache; and

[0131] an adjustment module **508**, configured to: when the elimination frequency parameter exceeds a first threshold, adjust the elimination ratio R to an elimination ratio R**1**, where the elimination ratio R**1** is greater than the elimination ratio R.

[0132] The adjustment module **508** is further configured to: when the elimination frequency parameter is less than a second threshold, adjust the elimination ratio R to an elimination ratio R**2**, where the elimination ratio R**2** is less than the elimination ratio R, and the second threshold is less than the first threshold.

[0133] The cache controller in this embodiment further includes:

[0134] The monitoring module **507** is configured to monitor an access frequency count of a cache line that belongs to the replacement set.

[0135] The elimination module **503** is configured to eliminate, from the replacement set, a cache line whose access frequency count is greater than a third threshold within a preset time period.

[0136] In this embodiment, the obtaining module **501** obtains the operation instruction. The operation instruction carries the destination address, and the destination address is the address that is to be accessed in the operation instruction. When the destination address hits no cache line cache line in the cache of the computer system, and the cache includes no idle cache line, the first selection module **502** selects the to-be-replaced cache line from the replacement set. The replacement set includes at least two cache lines. The elimination module **503** eliminates the to-be-replaced cache line from the cache. The storage module **504** stores, in the cache, the cache line obtained from the destination address. Therefore, in a cache line replacement process, the selection module **502** only needs to select the to-be-replaced cache line from the replacement set. The replacement set is preselected, and this effectively improves cache line replacement efficiency.

[0137] The foregoing describes the cache controller in this embodiment of the present disclosure from a perspective of a modular function entity, and the following describes a computer system in an embodiment of the present disclosure. Referring to FIG. **6**, an embodiment of the computer system in this embodiment of the present disclosure includes a processor **601**, a cache **602**, a cache controller **603**, and a memory **604**. The cache **602** is configured to cache some data in the memory **604**.

[0138] In some embodiments of the present disclosure, the processor **601** is configured to send an operation instruction; and

[0139] the cache controller **603** is configured to:

[0140] obtain the operation instruction, where the operation instruction carries a destination address, and the destination address is an address that is to be accessed in the operation instruction;

[0141] when the destination address hits no cache line cache line in the cache **602** of the computer system, and the cache includes no idle cache line, select a to-be-replaced cache line from a replacement set, where the replacement set includes at least two cache lines;

[0142] eliminate the to-be-replaced cache line from the cache; and

[0143] store, in the cache **602**, a cache line obtained from the destination address.

[0144] In some embodiments of the present disclosure, the cache controller **603** is further configured to perform the following steps:

[0145] determining multiple access frequency segments according to an access frequency count of each cache line in the cache and a preset division policy, where each access frequency segment is corresponding to a different access frequency count range;

[0146] determining, according to the access frequency count of each cache line in the cache, an access frequency segment to which each cache line belongs; and

[0147] selecting, from cache lines corresponding to the multiple access frequency segments, a to-be-replaced cache line according to a quantity M of to-be-replaced cache lines, to obtain the replacement set, where M is an integer not less than 2.

[0148] In some embodiments of the present disclosure, the cache controller **603** is further configured to perform the following step:

[0149] successively selecting, in ascending order of access frequency count ranges corresponding to the multiple access frequency segments, a cache line that belongs to each access frequency segment until a quantity of selected cache lines is equal to M.

[0150] In some embodiments of the present disclosure, the cache controller **603** is further configured to perform the following steps:

[0151] monitoring an elimination frequency parameter, where the elimination frequency parameter includes at least one of a miss rate of each cache line in the cache or a traversal frequency of each cache line in the cache; and

[0152] when the elimination frequency parameter exceeds a first threshold, adjusting an elimination ratio R to an elimination ratio R**1**, where the elimination ratio R**1** is greater than the elimination ratio R; or when the elimination frequency parameter is less than a second threshold, adjusting an elimination ratio R to an elimination ratio R**2**, where the elimination ratio R**2** is less than the elimination ratio R, and the second threshold is less than a first threshold.

[0153] In some embodiments of the present disclosure, the cache controller **603** is further configured to perform the following steps:

[0154] monitoring an access frequency count of a cache line that belongs to the replacement set; and

[0155] eliminating, from the replacement set, a cache line whose access frequency count is greater than a third threshold within a preset time period.

[0156] It may be clearly understood by a person skilled in the art that, for the purpose of convenient and brief description, for a detailed working process of the foregoing system, apparatus, and unit, reference may be made to a corresponding process in the foregoing method embodiments, and details are not described herein again.

[0157] An embodiment of the present disclosure further provides a computer program product for implementing an access request processing method, including a computer readable storage medium that stores program code, where an instruction included in the program code is used to execute the method procedure described in any one of the foregoing method embodiments. An ordinary person skilled in the art may understand that the foregoing storage medium may include any non-transitory machine-readable medium

capable of storing program code, such as a USB flash drive, a removable hard disk, a magnetic disk, an optical disc, a random-access memory (RAM), a solid state disk (SSD), or another non-volatile memory.

[0158] It should be noted that the embodiments provided in this application are merely examples. A person skilled in the art may clearly know that, for convenience and conciseness of description, in the foregoing embodiments, the embodiments emphasize different aspects, and for a part not described in detail in one embodiment, reference may be made to relevant description of another embodiment. The embodiments of the present disclosure, claims, and features disclosed in the accompanying drawings may exist independently, or exist in a combination. Features described in a hardware form in the embodiments of the present disclosure may be executed by software, and vice versa. This is not limited herein.

What is claimed is:

1. A cache management method applied to a computer system, the method comprising:

obtaining, by a cache controller, an operation instruction that carries a destination address, wherein the destination address is an address to be accessed in the operation instruction;

determining, by the cache controller, the destination address hits no cache line in a cache of the computer system, and wherein the cache comprises no idle cache line;

selecting, by the cache controller, a to-be-replaced cache line from a replacement set, wherein the replacement set comprises at least two cache lines;

eliminating, by the cache controller, the to-be-replaced cache line from the cache; and

storing, by the cache controller in the cache, a cache line obtained according to the destination address.

2. The cache management method according to claim **1**, further comprising:

determining, by the cache controller, multiple access frequency segments according to an access frequency count of each cache line in the cache and a preset division policy, wherein each access frequency segment is corresponding to a different access frequency count range;

determining, by the cache controller according to the access frequency count of each cache line in the cache, an access frequency segment to which each cache line belongs; and

selecting, by the cache controller from cache lines corresponding to the multiple access frequency segments, a to-be-replaced cache line according to a quantity M of to-be-replaced cache lines, to obtain the replacement set, wherein M is an integer not less than 2.

3. The cache management method according to claim **2**, wherein selecting, by the cache controller from cache lines corresponding to the multiple access frequency segments, a to-be-replaced cache line according to a quantity M of to-be-replaced cache lines comprises:

successively selecting, by the cache controller in ascending order of access frequency count ranges corresponding to the multiple access frequency segments, a cache line that belongs to each access frequency segment until a quantity of selected cache lines is equal to M.

4. The cache management method according to claim **3**, wherein:

the quantity M of to-be-replaced cache lines is determined according to an elimination ratio R and a total quantity of cache lines in the cache, wherein M is a product of the elimination ratio R and the total quantity of cache lines; and

the method further comprises:

monitoring, by the cache controller, an elimination frequency parameter, wherein the elimination frequency parameter comprises at least one of a miss rate of each cache line in the cache or a traversal frequency of each cache line in the cache,

determining, by the cache controller, the elimination frequency parameter exceeds a first threshold, and

adjusting, by the cache controller, the elimination ratio R to an elimination ratio R**1**, wherein the elimination ratio R**1** is greater than the elimination ratio R.

5. The cache management method according to claim **3**, wherein:

the quantity M of to-be-replaced cache lines is determined according to an elimination ratio R and a total quantity of cache lines in the cache, wherein M is a product of the elimination ratio R and the total quantity of cache lines; and

the method further comprises:

monitoring, by the cache controller, an elimination frequency parameter, wherein the elimination frequency parameter comprises at least one of a miss rate of each cache line in the cache or a traversal frequency of each cache line in the cache,

determining, by the cache controller, the elimination frequency parameter is less than a second threshold, and

adjusting, by the cache controller, the elimination ratio R to an elimination ratio R**2**, wherein the elimination ratio R**2** is less than the elimination ratio R.

6. The cache management method according to claim **3**, wherein the access frequency count of each cache line is obtained according to an access count of each cache line.

7. The cache management method according to claim **3**, wherein the access frequency count of each cache line is obtained according to an access count and access time of each cache line.

8. The method according to claim **4**, further comprising:

monitoring, by the cache controller, an access frequency count of a cache line that belongs to the replacement set; and

eliminating, by the cache controller from the replacement set, a cache line whose access frequency count is greater than a third threshold within a preset time period.

9. A computer system, comprising:

a cache;

a processor configured to send an operation instruction; and

a cache controller configured to:

obtain the operation instruction that carries a destination address, wherein the destination address is an address to be accessed in the operation instruction,

determine the destination address hits no cache line in the cache of the computer system, and the cache comprises no idle cache line,

select a to-be-replaced cache line from a replacement set, wherein the replacement set comprises at least two cache lines,

eliminate the to-be-replaced cache line from the cache, and

store, in the cache, a cache line obtained from the destination address.

10. The computer system according to claim **9**, wherein the cache controller is further configured to:

determine multiple access frequency segments according to an access frequency count of each cache line in the cache and a preset division policy, wherein each access frequency segment is corresponding to a different access frequency count range;

determine, according to the access frequency count of each cache line in the cache, an access frequency segment to which each cache line belongs; and

select, from cache lines corresponding to the multiple access frequency segments, a to-be-replaced cache line according to a quantity M of to-be-replaced cache lines, to obtain the replacement set, wherein M is an integer not less than 2.

11. The computer system according to claim **10**, wherein the cache controller is further configured to:

successively select, in ascending order of access frequency count ranges corresponding to the multiple access frequency segments, a cache line that belongs to each access frequency segment until a quantity of selected cache lines is equal to M.

12. The computer system according to claim **11**, wherein the cache controller is further configured to:

monitor an elimination frequency parameter, wherein the elimination frequency parameter comprises at least one of a miss rate of each cache line in the cache or a traversal frequency of each cache line in the cache;

determine the elimination frequency parameter exceeds a first threshold; and

adjust an elimination ratio R to an elimination ratio R**1**, wherein the elimination ratio R**1** is greater than the elimination ratio R.

13. The computer system according to claim **11**, wherein the cache controller is further configured to:

monitor an elimination frequency parameter, wherein the elimination frequency parameter comprises at least one of a miss rate of each cache line in the cache or a traversal frequency of each cache line in the cache;

determine the elimination frequency parameter is less than a second threshold; and

adjust the elimination ratio R to an elimination ratio R**2**, wherein the elimination ratio R**2** is less than the elimination ratio R.

14. The computer system according to claim **11**, wherein the cache controller is further configured to:

monitor an access frequency count of a cache line that belongs to the replacement set; and

eliminate, from the replacement set, a cache line whose access frequency count is greater than a third threshold within a preset time period.

15. The computer system according to claim **10**, wherein the cache controller is further configured to:

obtain the access frequency count of each cache line according to an access count of each cache line.

16. The computer system according to claim **10**, wherein the cache controller is further configured to:

obtain the access frequency count of each cache line according to an access count and access time of each cache line.

17. A cache controller, comprising:

a processor; and

a memory comprising a plurality of computer-executable instructions stored thereon which, when executed by the processor, cause the cache controller to:

obtain an operation instruction that carries a destination address, and wherein the destination address is an address to be accessed in the operation instruction,

determine the destination address hits no cache line in a cache, and the cache comprises no idle cache line,

select a to-be-replaced cache line from a replacement set, wherein the replacement set comprises at least two cache lines,

eliminate the to-be-replaced cache line from the cache, and

store, in the cache, a cache line obtained from the destination address.

18. The cache controller according to claim **17**, wherein the computer-executable instructions when executed by the processor, further cause the cache controller to:

determine multiple access frequency segments according to an access frequency count of each cache line in the cache and a preset division policy, wherein each access frequency segment is corresponding to a different access frequency count range;

determine, according to the access frequency count of each cache line in the cache, an access frequency segment to which each cache line belongs; and

select, from cache lines corresponding to the multiple access frequency segments, a to-be-replaced cache line according to a quantity M of to-be-replaced cache lines, to obtain the replacement set, wherein M is an integer not less than 2.

19. The cache controller according to claim **18**, wherein the computer-executable instructions when executed by the processor, further cause the cache controller to:

successively select, in ascending order of access frequency count ranges corresponding to the multiple access frequency segments, a cache line that belongs to each access frequency segment until a quantity of selected cache lines is equal to M.

20. The cache controller according to claim **19**, wherein the computer-executable instructions when executed by the processor, further cause the cache controller to:

monitor an elimination frequency parameter, wherein the elimination frequency parameter comprises at least one of a miss rate of each cache line in the cache or a traversal frequency of each cache line in the cache;

determine the elimination frequency parameter exceeds a first threshold; and

adjust an elimination ratio R to an elimination ratio R**1**, wherein the elimination ratio R**1** is greater than the elimination ratio R.

* * * * *